Year: 2015

A Framework for a Cost-Efficient Cloud Ecosystem

Salant, Eliot ; Leitner, Philipp ; Wallbom, Karl ; Athes, James

Originally published at:
Salant, Eliot; Leitner, Philipp; Wallbom, Karl; Athes, James (2015). A Framework for a Cost-Efficient Cloud Ecosystem. In: eChallenges e-2015 Conference, Vilnius, Lithuania, 25 November 2015 - 26 November 2015, s.n..

A Framework for a Cost-Efficient Cloud Ecosystem

Eliot Salant[1], Philipp Leitner[2], Karl Wallbom[3], James Ahtes[4]
[1]IBM Research Haifa, University Campus, Haifa, 31905, Israel
Tel: +972-48296121, Fax: +972-48296114, salant@il.ibm.com
[2]Software evolution & architecture lab, University of Zurich, Switzerland
Tel: +41 44 63 545 81, leitner@ifi.uzh.ch
[3]Cloudmore, Sweden
Tel: +46 8 420 130 40, karl.wallbom@cloudmore.com
[4]Atos, Avenue Diagonal 200, Barcelona, 08018, Spain
Tel: + 34 914 569 841, james.ahtes@atos.net

Abstract: Cloud Computing has developed rapidly over the last ten years with world-wide spending on public and private cloud hosting passing the $32bn. mark this year, and savings to businesses using the cloud are typically stated as better than 30% due to the cloud's ability to better take advantage of economies of scale. Yet, in actuality, in commercial data centres the utilization of resources still remain low. In this paper we introduce a new approach from CloudWave which uses user-defined performance goals as well as monitoring across the Cloud ecosystem as a means for guiding runtime adaptation of both the cloud infrastructure and the hosted service to improve overall performance. In addition, inspired by the current industry trend of *DevOps* methodology, CloudWave feeds runtime performance and monitoring insights back to the DevOps engineers to enable identification and refactoring of problematic code sections. The potential impact for improving data centre efficiency is huge – as Google researcher John Wilkes has pointed out that at such large scales, "A few percent here, a few percent there, and all of the sudden, you're talking about huge amounts of money". After two years of research, the CloudWave platform is starting to test its ideas on actual applications, which in turn is exposing new avenues for research.

1. Introduction

Cloud computing scarcely needs an introduction – a recent Goldman Sachs [1] study predicts a 30% increase in enterprise spending levels on cloud computing infrastructures and platforms over the next three years. Similarly, global Software as a Service (SaaS) revenues are expected to enjoy double digit percentage increases over this same time period.

A main driver for cloud adoption is, of course, financial – a Booz Allen Hamilton study [2] from 2009 predicted that transition to a virtualized cloud computing approach from a traditional data centre could save a 1,000 server deployment 50-67% of the total lifecycle costs.

Financial savings from moving to a cloud environment are commonly attributed to factors due to economies of scale which are obtained by dynamically being able to pool resources and handle multi-tenancy hosting of customers as well as obtain higher utilization of physical servers. While this has proven true, there is still much more research required for a Cloud environment to reach its full utilization potential; the average aggregated CPU utilization in a commercial data centre is below 20% [3].

Cloud management frameworks of today such as Apache Mesos [4] are typically *reservation-centric*, where resources are assigned to a workload based on an estimation of how much capacity will be required. However, estimating required resources is hard; an analysis of a production cluster at Twitter shows that most workloads overestimate their requirements by a factor of 10 [3], leading to low server utilization. Infrastructure as a Service (IaaS) systems, such as OpenStack or EC2, based on user request, typically allot virtual resources of a given *flavour* which gives a template allocation for resources such as virtual number of cores, RAM, disk size, bandwidth etc.

CloudWave, a three year European Union sponsored FP7 project, has been examining increasing application performance and data centre efficiency by taking a new approach to cloud management and cloud software development that is inspired by DevOps concepts. The CloudWave approach defines user-specified performance goals, and through utilizing comprehensive monitoring of the cloud ecosystem, modifies both the infrastructure and the application to meeting these goals. In addition to these operation-side issues, software development for the cloud is also known to be difficult [5]. As cloud systems by definition are black boxes to developers, building cost-efficient and resilient applications is hard. CloudWave also takes advantage of the deep, stack level monitoring data collected for adaptation analysis to feed back information to DevOps engineers to help guide a refactoring of their code to better meet actual production runtime conditions.

This paper will introduce the main concepts behind CloudWave, identify the main stakeholders of CloudWave, and finally show the business potential they represent for these stakeholders. Further, insight from a commercial use case provider will be highlighted.

2. Objectives

The main goal of CloudWave is to use the principles of DevOps – the convergence of Cloud-based software development and operations – to dynamically increase the efficiency of a hosted service through both adaptation of the Cloud ecosystem, as well as by supplying DevOps engineers with the feedback required to refactor their application.

A DevOps engineer is able to specify the performance goals of the service, and can use a system-supplied toolkit to enable monitoring of these performance goals. Performance goals can either be defined at the level of measurable infrastructure metrics (e.g. CPU load, memory limits, etc.), similar to the way that commercial systems specify auto-scale thresholds, or can be more complicated combinations of any number of metrics. CloudWave also supplies an API and library for the creation of software probes within the application itself, which will allow for monitoring of application-side metrics, such as experienced delay.

Adaptation of the system is orchestrated by an Adaptation Engine, which is a collection of sub-engines which make weighed recommendations for adaptation based on their particular field of knowledge. In the simplest case, the Adaptation Engine will trigger an adaptation of the infrastructure, adding or migrating virtual resources. In the case where the hosted service has been written to be CloudWave-compliant, a *coordinated* adaptation consisting both of application and infrastructure configuration can be executed.

This concept is validated by a reference implementation which is tested against actual use cases from industry. A business case for adopting CloudWave-like technology has been developed.

3. Methodology

CloudWave aims to develop a modular, open reference. The abstraction level of the interfaces is meant to be high enough to allow third party and legacy components to be integrated in with the general architecture. CloudWave is not creating a new cloud infrastructure – the reference architecture is built on top of OpenStack - but rather presents a framework for the orchestration of services and cloud infrastructure using the native methods that they offer.

In parallel, utilizing the same monitoring and runtime analysis infrastructure, CloudWave offers a toolkit which aids developers in both modifying their services to improve performance from observed workload behaviour, as well as shorten the development cycle for adding new features.

The CloudWave project itself has been utilizing the principles of Agile in its development, re-evaluating design principles based on cycles of design, implement and validate against use cases brought in by its industrial partners. Over time, the use cases grow more sophisticated as use case partners realize the potential the CloudWave has to offer, and the CloudWave architecture grows more mature to meet the evolved use cases.

4. Technology Description

As shown in Figure 1, the CloudWave architecture is built around three main conceptual components, Execution Analytics, Coordinated Adaptation, and Feedback Driven Development (FDD). The CloudWave Infrastructure, depicted by the arrows, provides the communication between the components and the supporting services. These components will now be described at a high level.

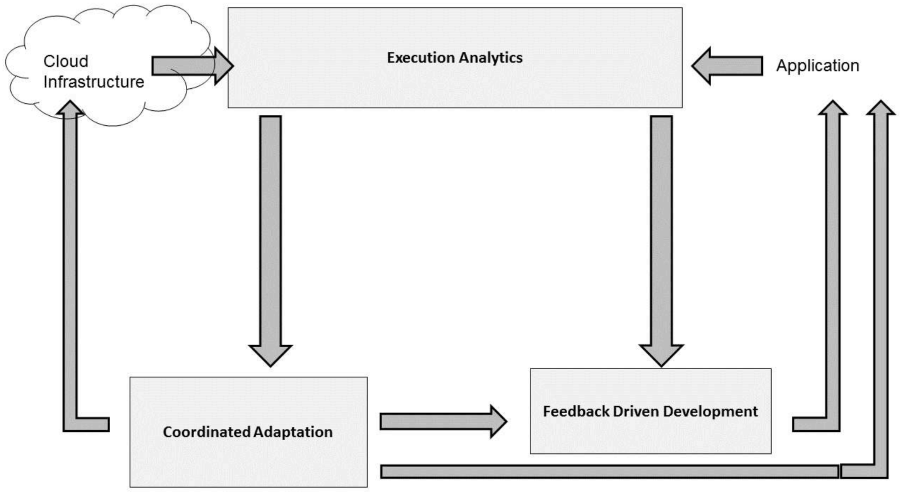

Figure 1: The CloudWave Conceptual Architecture

Execution Analytics

The ability of CloudWave to cause the cloud ecosystem to adapt for higher efficiency depends on being able to collect performance metric data from all levels of the ecosystem, namely:

- Physical level of the cloud infrastructure (e.g. network and server parameters).
- Virtual level such as virtual machines (VMs) and virtual network segments.
- Application level, through log files written by the application, or monitoring probes.
- System level, through log files written by the system.

Information collected must be persisted for potentially long periods of time to allow for analytics on historical performance data. As there potentially might be extremely large amounts of monitoring data flowing into CloudWave from a large number of applications on nodes across the data centre, Execution Analytics must also be able to supply a scalable solution for local filtering and compressing collected data.

The trigger conditions for invoking adaptation of system behaviour in CloudWave are set through the definition of *enactment points* which define a threshold limitation for one or more CloudWave monitored metrics. Typically this threshold value is set to indicate that an application is approaching the point where performance will drop below an acceptable value.

It is the responsibility of the Execution Analytics component to not just store the definition of the enactment points, but to also monitor and alert the Coordinated Adaptation component when the enactment point has been triggered.

Coordinated Adaptation

The concept of Coordinated Adaptation in CloudWave represents an orchestrated modification of either the infrastructure configuration, application runtime behaviour, or both. Invoked when an enactment point is triggered, the Adaptation Engine receives input regarding the enactment point (i.e. the reason for requesting the adaptation), access to historical metric data to enable machine learning capabilities, information about system and application configuration, constraints and Service Level Agreement (SLA) requirements.

The Adaptation Engine has a known number of adaptation actions that it can request from both the infrastructure and the application. Infrastructure adaptation actions affect the (virtual) resources allocated to a service, whereas application adaptation actions typically affect the behaviour of the CloudWave-compatible application. An initial evaluation of potential adaptation actions against a dynamic set of constraints (e.g. SLA specified hosting budget constraints, application deployment constraints etc.) can initially eliminate a number of candidate adaptation actions. Individual adaptation sub-engines then attempt to recommend an adaptation solution, based on their specific area of focus. Finally, the Adaptation Engine, using a predefined weighing scheme, will consider all recommended actions from the various sub-engines and generate an adaptation plan of action which is then passed to the CloudWave Infrastructure for execution.

Feedback Driven Development (FDD)

One of the main concepts behind DevOps includes promoting closer collaboration between the engineers involved in the development of a service and actual operations involved in deployment and actual workload behaviour of the service. Additionally, DevOps advocates agility, where new versions of a service are released frequently with incremental changes to its functionality.

FDD in CloudWave is meant to support these concepts by harnessing the holistic view of a service ecosystem supplied by the Execution Analytics component. FDD presents a framework to the DevOps engineer which features a plug-able suite of tools to help improve application performance or new features, as well as provide insights into actual production workload behaviour. Additionally, through the FDD, DevOps engineers are able to couple insights from FDD's view of the service ecosystem with their deep knowledge of their service to raise calls for system action through the definition of enactment points.

5. Results

The first year of CloudWave was primarily dedicated to implementing an initial version of the CloudWave infrastructure built on top of OpenStack. Ceilometer, supplied with OpenStack, was used for collecting monitoring information from the virtual machines. Esper, (http://www.espertech.com/esper/) was used for complex event processing by the Execution Analytics component.

On top of these facilities, the project has introduced the notion of 3-D monitoring [6]. 3-D monitoring is about integrating cloud monitoring data from the physical layer (physical servers), virtualization layer (virtual machines), application architecture layer (service compositions and application topologies), and application business logic layer (individual services and applications).

One use case for 3-D monitoring in the project is runtime adaptation of cloud applications running on top of the CloudWave stack. Triggered by enactment point violation defined across multiple layers, infrastructure adaptation actions include scale-up, scale-out and virtual machine migration. Additionally, the ability to notify an application and have it adapt was also demonstrated.

Additionally, we have shown that 3-D monitoring can be of tremendous value to software developers as well [7]. This builds on the current trend of Application Performance Management (APM), as implemented by tools such as New Relic (http://newrelic.com). However, CloudWave Feedback Driven Development goes beyond the state of the art in APM, as it gives developers access to monitoring data beyond their single application.

6. Business Benefits

While the CloudWave approach potentially delivers value to a large number of stakeholders, we have identified three groups as the main beneficiaries of the project:

Cloud Service Providers, and specifically the operators of cloud infrastructures, currently struggle with optimally using their available computing resources. As the applications deployed on their infrastructures are black boxes to them, it is difficult to allocate resources optimally and maximize data centre utilization. CloudWave supports Cloud Infrastructure Operators by providing monitoring and adaptation capabilities not only on infrastructure level, but also on application topology and application level. The result decreases their operational expenses (OpEx) while making them a more attractive, competitive option to their cloud application service provider customers.

On the *Cloud Service Consumer* side, Application Software Developers can use monitoring data from the infrastructure (integrated with Application Performance Monitoring data) to build better (i.e., better performing, more resilient, and more cost-efficient) cloud applications, and to detect potential quality regressions (e.g., performance faults) prior to deploying to production. This also provides substantial value to operators and DevOps engineers on the cloud consumer side, as this allows them to improve the stability of production deployments and reduce downtimes and episodes of degraded customer experience due to the rollout of performance-critical bugs. The result means better quality applications, quicker time-to-market, and agile, iterative release cycles in a more integrated system with their IaaS hosts than current monitoring and DevOps solutions provide.

In the following, we detail the potential business value that CloudWave brings to each of these stakeholders in more detail.

6.1 Benefits for the Infrastructure Provider

Increased backend efficiency for infrastructure cloud providers' data centers is extremely important to the business's bottom line. Google found that an extra 0.5 seconds in search page generation time due to returning more results, dropped traffic by 20% - which translates into millions of dollars in lost revenue [8]. Such optimization through better monitoring and management of their infrastructure also leads to financial gains in decreased operational costs (OpEx) and better use of current hardware (CapEx).

Commercial cloud providers have been locked in a pricing war for a number of years, with annual reductions in rates offered to customers typically dropping more than 20% a year [9]. As the commoditization of cloud computing continues, providers look for innovative tools and services to offer on top of their base IaaS offering, creating new revenue sources and end-to-end offerings that can further differentiate them from the competition. CloudWave's advances in monitoring and DevOps transcend to the SaaS layer, as well, making them more attractive to their Application Service Provider and developer customers.

6.2 Benefits for Application Developers

Agile methodologies, with DevOps tools and culture in particular, are seeing increased adoption in software development [5]. CloudWave builds on top DevOps solutions, with close integration to 3-D monitoring and to the adaptation framework of the platform. The holistic approach creates a built-for-DevOps infrastructure that uses such agile development, monitoring and automation to deliver rapid software innovation, yet cost effective for both the developer (SaaS) and hosting cloud provider (IaaS).

Additionally, by now it is well-understood that delivering broken, or even badly performing, software releases to production is expensive for application providers. For instance, AppDynamics has recently released a report claiming that for Fortune 1000 companies, the average aggregated costs of one hour of downtime of a business-critical application (e.g., because a broken build had to be rolled back) was between 500.000 and 1.000.000 US dollars [1]. Other studies claim that close to 50% of users of Web shops tend to abandon purchases on sites that take more than 2 seconds to load [10]. Consequently, large IT-intensive businesses such as Facebook, invest heavily into detecting performance regressions or broken builds before deploying them to production [11].

The CloudWave FDD approach helps prevent such costly performance regressions. Using sophisticated prediction models and visualizations, application developers are warned already at compilation time, before deploying badly performing application code to production. Further, 3-D monitoring gives developers insight into how the code they are writing relates to cloud resource costs (e.g., for virtual machines required to execute their applications and services), and allows them to better trade off potential revenues of new features with the additional costs that the implementation of these features would lead to. FDD uses customizable feedback for improved agile development, resulting in quicker time-to-market, shortened maintenance cycles and more reliable cloud applications for a SaaS service provider and its end-user customers.

6.3 Benefits for Application Operators

With the increasing popularity of DevOps practices there has been a huge development in supporting tools [12] for everything from configuration management to test and build systems. However, even with these tools, the impact of new architectures, such as microservices, is not always evident to developers or even operations. If an application experiences performance issues it might be caused by the underlying infrastructure outside of the application operator's control [13]. Architectural changes or adaptations of the application might not solve the problem or would incur unnecessary costs for the application owner. They might lack the required infrastructure metrics or capabilities to properly analyse the problem since the infrastructure is often a black box this information is usually unavailable [14].

The CloudWave Coordinated Adaptation can help redeem application issues affected by the infrastructure before they occur by invoking changes to the infrastructure or adapt the application with pre-specified adaptation responses. The 3-D monitoring offers application operators a holistic view to previously unavailable data. An application operator can for example include adaptations to enforce application SLA or deployment constraints.

7. Use Case Study

CloudWave has developed a VoIP application meant for hosting in a private cloud. The application is a communication tool that is provided free of charge but has a premium subscription that gives benefits to paying users. Use of the CloudWave Framework extends the underlying OpenStack cloud to maximize efficiency for both its application and platform. The application has high bandwidth requirements, however due to economic constraints, the amount of resources that can be allocated to the application is capped.

The application owner configures an application SLA in the CloudWave DevOps Portal, which is the central point of access to the CloudWave system. He defines enactment points and configures the infrastructure monitoring. The application is then deployed and users start joining. As service use increases, an enactment point is triggered, indicating that the application might soon run into performance problems. The Adaptation Engine determines that a scale-out adaptation will likely fix the problem and initiates the adaptation response.

After a while, more and more users start to use the video chat functionality of the application. This in turn dramatically increases network load, which will eventually lead to performance problems. The Adaptation Engine determines that there is no infrastructure adaptation that can redeem the situation without breaking the economic constraints so it requests an adaptation from the application. It weighs the different adaptation responses and determines the best solution is to disable the video chat feature for non-paying customers.

The application owner decides to refactor his application since the video chat feature is an integral part of the user experience. FDD analysis helps the application developer identify portions in the code which may impact performance, guiding the code refactoring. Upon modification, the improvement in application behaviour is verified in a sandbox environment and then deployed in production.

8. Progress

CloudWave is now approaching the end of its second year. Knowledge gained as the project progresses has resulted in a number of changes to the architectural approach, as well as pointed out directions for research directions.

In particular, after the first year we realized that the concept of an Adaptation Engine capable of determining the proper application and infrastructure adaptations based on machine learning from past system behaviour is at best impractical, if not impossible, due to the large state of possible system configurations. In addition, this type of approach overlooks financial constraints that might be imposed on the allotment of additional resources. Instead, a more promising direction for the Adaptation Engine has been breaking this into a number of pluggable sub-engines, all of which make adaptation recommendations based on one area of specialty. Constraint engines restrict the options available to these sub-engines – for example the Cost Engine will not allow scale-out if the allocation of a new virtual machine is expected to exceed the budget for hosting the service as specified in its SLA. The recommendations of the individual engines are assigned a weight (typically by a DevOps engineer) and therefore the chosen adaptation action will result for a weighted scoring of all the sub-engines.

Close collaboration with the industrial use case partners has emphasized the need to support closed, legacy applications that will not support recompilation with a monitoring probe. Instead, the CloudWave architecture was adapted to extract whatever logged information such applications might provide, and inject the information into the CloudWave monitoring bus as if it were provided by a probe.

It is expected that by the end of the project's third and final year the industrial partners in the project will have benchmarked the performance of their applications both with and without CloudWave in a realistic environment.

9. Related Research Efforts

CloudWave is embedded in the fruitful environment of European cloud computing related research and innovation projects. CloudWave collaborates with a number of related projects as part of the *Software Engineering for Services and Applications* project cluster (https://eucloudclusters.wordpress.com/software-engineering-for-services-and-applications/), as well as within the *New Approaches for Infrastructure Services* cluster (https://eucloudclusters.wordpress.com/new-approaches-for-infrastructure-services/).

Particularly, the project has strong ties to the SeaClouds project [15]. SeaClouds works on the monitoring, management and composition of cloud services based on standards, such as OASIS TOSCA or CAMP. SeaClouds and CloudWave are organizing a joint workshop as part of the Fourth European Conference on Service-Oriented and Cloud Computing (ESOCC).

Another related project is MODAClouds [16], which investigates a model-driven approach for designing and executing cloud applications. MODAClouds is geared towards supporting the cloud user, and has a strong focus on modelling applications independently of any concrete provider.

This is similar to the HyVar project (http://www.hyvar-project.eu/hyvar/), a newly launched European project funded via the H2020 initiative. HyVar models cloud applications using the ideas of software product lines. The project has only recently started, and little concrete information is currently publicly available.

Another recently launched related initiative is the ENTICE project (http://www.entice-project.eu). ENTICE proposes to provide a registry of virtual machine images and templates for different use cases and different cloud systems.

Finally, the ARTIST project [17] deals with the migration of legacy applications to the cloud. This is an aspect that we also touch upon in CloudWave, even though the primary use case for

CloudWave is on cloud-native applications, that is, applications that have never been intended to be executed in an on-premise setting.

10. Conclusions

It has been clear as the project has developed that a sematic understanding of the Cloud-side portion of the application is required to make proper adaptation decisions – for example, understanding that a virtual machine is running a database with huge amounts of data would most likely preclude the migration of this VM to different physical host. Additionally, for the best adaptation decisions, we are seeing the need to be able to understand the physical hardware underneath the Cloud virtualization layer.

CloudWave's third and final year efforts will include examining ways of modelling both the data centre resources and the deployed application such that this information will be both available to Adaptation Engine as well as extendable with meta-data.

All of CloudWave's architectural and scientific documents are being made publically available on its web site (www.cloudwave-fp7.eu) as well as a number of its analytical tools for feedback driven development.

References

[1] Forbes, "Roundup of Cloud Computing Forecasts and Market Estimates, 2015," 24 1 2015. [Online]. Available: http://www.forbes.com/sites/louiscolumbus/2015/01/24/roundup-of-cloud-computing-forecasts-and-market-estimates-2015/.
[2] Booz Allen, "The Economics of Cloud Computing," October 2009. [Online]. Available: http://www.boozallen.com/insights/2009/10/42656904.
[3] C. Delimitrou and C. Kozyrakis, "Quasar: Resource-Efficient and QoS-Aware Cluster Management," in 19th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), 2014.
[4] Mesos, [Online]. Available: http://mesos.apache.org/documentation/latest/mesos-architecture/.
[5] J. Cito, P. Leitner, T. Fritz and H. Gall, "The Making of Cloud Applications – An Empirical Study on Software Development for the Cloud," in Proceedings of the 10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT International Symposium on Foundations of Software Engineering (ESEC/FSE), New York, 2015.
[6] C. C. Marquezan, D. Bruneo, F. Longo, F. Wessling, A. Metzger and A. Puliafito, "3-D cloud monitoring: Enabling effective cloud infrastructure and application management.," in 10th International Conference on Network and Service Management (CNSM), 2014.
[7] J. Cito, P. Leitner, H. Gall, A. Dadashi and A. Roth, "Runtime Metric Meets Developer – Building Better Cloud Applications Using Feedback," in Proceedings of the 2015 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming & Software, New York, 2015.
[8] Gilden, 2006. [Online]. Available: http://glinden.blogspot.com/2006/11/marissa-mayer-at-web-20.html.
[9] Rightscale, 2013. [Online]. Available: http://www.rightscale.com/blog/cloud-industry-insights/cloud-price-reductions-definitive-analysis-2013-trends.

[10] Kissmetrics, [Online]. Available: https://blog.kissmetrics.com/.
[11] E. Bakshy and E. Frachtenberg, “Design and Analysis of Benchmarking Experiments for Distributed Internet Services,” in World Wide Web Conference, 2015.
[12] Profitbricks, [Online]. Available: https://blog.profitbricks.com/51-best-devops-tools-for-devops-engineers/.
[13] P. Leitner, J. Cito, “Patterns in the Chaos - a Study of Performance Variation and Predictability in Public IaaS Clouds,” arXiv:1411.2429 [cs.DC], 2015.
[14] S. Sáez, V. Andrikopoulos, F. Wessling and C. C. Marquezan, “Cloud Adaptation and Application (Re-)Distribution: Bridging the Two Perspectives.,” in Proceedings of EDOC Workshops, 2014.
[15] A. Brogi, A. Ibrahim, J. Soldani, J. Carrasco, J. Cubo, E. Pimentel, and F. D'Andria. “SeaClouds: a European project on seamless management of multi-cloud applications.” SIGSOFT Softw. Eng. Notes 39, 1, 1-4. 2014.
[16] D. Ardagna, E. Di Nitto, G. Casale, D. Petcu, P. Mohagheghi, S. Mosser, P. Matthews, A. Gericke, C. Ballagny, F. D'Andria, C. Nechifor, and C. Sheridan. “MODAClouds: a model-driven approach for the design and execution of applications on multiple clouds”. In Proceedings of the 4th International Workshop on Modeling in Software Engineering (MiSE '12). 2012.
[17] A. Bergmayr, H. Bruneliere, J. L. Canovas Izquierdo, J. Gorronogoitia, G. Kousiouris, D. Kyriazis, P. Langer, A. Menychtas, L. Orue-Echevarria, C. Pezuela, M. Wimmer, "Migrating Legacy Software to the Cloud with ARTIST," in Proceedings of the 17th European Conference on Software Maintenance and Reengineering (CSMR), 2013, pp.465-468, 2013